The Dendro research data management platform

Applying ontologies to long-term preservation in a collaborative environment

João Rocha da Silva
Faculdade de Engenharia da Universidade do Porto / INESC TEC
Portugal
joaorosilva@gmail.com

João Aguiar Castro
Faculdade de Engenharia da Universidade do Porto / INESC TEC
Portugal
joaoaguiarcastro@gmail.com

Cristina Ribeiro
DEI — Faculdade de Engenharia da Universidade do Porto / INESC TEC
Portugal
mcr@fe.up.pt

João Correia Lopes
DEI — Faculdade de Engenharia da Universidade do Porto / INESC TEC
Portugal
jlopes@fe.up.pt

ABSTRACT

It has been shown that data management should start as early as possible in the research workflow to minimize the risks of data loss. Given the large numbers of datasets produced every day, curators may be unable to describe them all, so researchers should take an active part in the process. However, since they are not data management experts, they must be provided with user-friendly but powerful tools to capture the context information necessary for others to interpret and reuse their datasets. In this paper, we present Dendro, a fully ontology-based collaborative platform for research data management. Its graph data model innovates in the sense that it allows domain-specific lightweight ontologies to be used in resource description, acting as a staging area for later deposit in long-term preservation solutions.

Categories and Subject Descriptors

H.3 [**Information Search and Retrieval**]: Online Information Services Data sharing

Keywords

Research data management, data curation, ontologies, data repositories, Dendro

1. INTRODUCTION

It is widely accepted that research data management should start as soon as possible in the research workflow. However, most research data management platforms like CKAN, Zenodo or Dryad are designed for publishing "finished" datasets that can be *cited*. This *a posteriori* data management timing yields very high-quality and highly-selected datasets, but in many cases the number of datasets that are actually published can be quite low. Empty dataset archives and repositories are still commonplace [Nelson 2009; Borgman 2012].

Several data management projects focus on supporting collaboration within research groups and making daily data management activities easier. The resulting tools are therefore entry points through which the datasets can enter a preservation workflow [Hodson 2011; Shotton 2012]. These solutions focus on providing easy-to-use shared storage spaces with regular automated backups, connected to a data repository. The main objectives were to capture data as early as possible and leave detailed description for later (*curation by addition*). In both cases, only a minimal set of metadata is required upon initial submission, leaving the decision to enrich the metadata to the researcher and/or curator.

Current data management platforms often limit the metadata that can be added to a dataset to generic descriptors (e.g. Dublin Core) or a pre-existent set of descriptors that depositors are asked to fill in at the time of deposit. CKAN [Open Knowledge Foundation 2014] is an exception, as it allows an additional set of arbitrary metadata to be added to deposited datasets, in the form of *ad-hoc* text fields. This allows domain-specific metadata to be recorded, although without any pre-defined meaning or standards-compliance.

Dendro, our proposed research data management platform, aims to establish a tradeoff between close proximity to the researcher, incremental data description, quick and simple deposit and no metadata requirements. It uses a triple store to support an ontology-based data model in order to satisfy the metadata needs of different research communities. No metadata requirements exist at the time of deposit, but the basic descriptors (creator, modification date, creation

iPres 2014 conference proceedings will be made available under a Creative Commons license. With the exception of any logos, emblems, trademarks or other nominated third-party images/text, this work is available for re-use under a Creative Commons Attribution 3.0 Unported license. Authorship of this work must be attributed. View a copy of this license.

date, etc.) are provided, and the user is expected to fill them in. Richer descriptors are presented as *recommendations* that researchers and curators can choose to fill in or not for each resource. From a preservation standpoint, it is completely supported by open-source software built for cloud-level scalability. Its underlying data model makes data easier to preserve due to its intrinsic readability and Linked Open Data foundation. Its dispenses a relational database and is designed to foster dataset integration in the Semantic Web as Linked Open Data (LOD). An interesting side-effect that stems from the adoption of this model is that the usual layers of relational-LOD translation logic that often exist in solutions that provide LOD compatibility solutions are eliminated. An practical example is Semantic MediaWiki, that uses a relational database in its transactional system and an RDF store for semantic querying, requiring specific code to maintain a permanent mapping between the two solutions. We argue that, by removing the dependency on a relational database altogether, we can remove the concerns over its migration when the system is rendered obsolete and provide an ontology-based metadata model from end to end.

2. A TRIPLE-BASED DATA MODEL

Unlike key-value metadata representations, a linked data representation gives structure and explicit *meaning* to metadata values, allowing datasets, papers, researchers and other research-related resources to be connected by meaningful links. These meanings can also be reused from existing specifications (*ontologies*) or newly created if no ontology defines them. The advantages of this representation from a preservation point of view include the simplicity of the data model and its superior flexibility (it can grow incrementally as more ontologies for different domains are designed). When registering the URI of the creator's web page in the `dc:creator` of a dataset, a system built on linked data will record the *meaning* of that string value, unlike a relational system, where there is no distinction between different types of values. These meanings are specified with ontologies, which can be shared along with the data and the metadata records. In a preservation environment, the advantages are clear: linked data provides great support for self-documented metadata which can also be represented in RDF format—an open, plain-text representation with minimal reliance on specific processing software.

Dendro was designed from the start as a user-friendly interface targeted at users without data management skills. As they interact with the system, a linked data knowledge base is built using ontologies in the background. It is similar to a semantic wiki in the sense that it allows users to collaboratively shape the underlying graph through their daily interaction and directly uses ontologies for parameterization (no mapping between a relational model and a triple store representation ever occurs). Moreover, Dendro's data model is built to offer programmers the appropriate granularity for descriptor-level analysis, allowing the easy combination of descriptors from several domains. We illustrate this by comparing Dendro's data model with the data model of Semantic MediaWiki, perhaps the most widely known semantic wiki.

2.1 Dendro vs. Semantic MediaWiki

Semantic MediaWiki (SMW) is built around ontologies that are used to give *meaning* to the links established between wiki pages (*semantic links*). It offers two different interfaces for establishing semantics between wiki pages. The first one is the standard text editor where semantics can be added to a link tag. For example, one can write: `The author of this paper was [[author:Bob].` In a wiki page. The result would be a very small wiki page with link to **Bob**'s page in the wiki. Internally, a link would be established between the page being edited and the web page of the author. To apply this technique to dataset description, one would start by creating a wiki page for each file in a dataset and write a plain text description containing several of these links. This way, semantic metadata could be embedded in the metadata descriptions.

Another alternative is using SMW's *semantic forms*. These are more structured interfaces designed for users to fill in a predefined set of links. However, these predefinitions have to be specified *a priori*; researchers cannot select descriptors to include in their metadata sheets, having instead to rely on a single template.

Our past work on DataNotes, an extension to SMW [Rocha da Silva et al. 2013] proposed a modification to the platform to allow researchers to freely include descriptors from several ontologies in their descriptions. Extensive changes had to be made to the business logic and user interface, but the issues caused by having a relational and a triple-based side by side still remained.

2.2 The advantages of a graph-based model

Ontologies and triple stores allow us to tackle the research data management challenge in a unique manner, enabling the representation of resources with different sets of attributes, even when they are not known at the time of modeling. Realizing the advantages of a graph-based data model over the constraints posed by a relational approach, a design for a multi-domain research data management system has proposed a similar ontology-based architecture built on triple stores [Li et al. 2013].

The data model behind Dendro has the right granularity for describing any kind of resource using variable descriptors without incurring in a convoluted relational database schema, which would mean complex queries and heavy JOIN operations every time we wanted to access the descriptors of a resource. Also, since the core data model of the platform uses a triple store, it becomes possible to directly load ontologies from different domains into the knowledge base and reuse the concepts specified in those ontologies. This allows domain experts to specify their own ontology using high-level tools like Protégé[1] (or just reuse existing ones) and load them into Dendro, thus enabling the new concepts to be used in the description of research data assets. Given the open nature of ontologies and their asynchronous evolution through reuse, platforms like Dendro can retain a higher level of interoperability than conventional RDB-based ones. With this approach we plan for obsolescence in a positive way: the data more easily survive the obsolescence of the Dendro platform, as the contents of the entire data model can be exported as Linked Open Data (LOD). The data model itself will also be public and self-documented, since

[1] `http://protege.stanford.edu`

it is good practice of ontology design to document ontology concepts at design time, via the common `rdfs:label` and `rdfs:comment` description properties—information that is also used by Dendro in its user interfaces.

2.3 Dendro in the preservation workflow

Figure 2 shows Dendro's role in the research data management ecosystem as it supports the process at different points in time.

1. Data creation, description and sharing within the research group throughout their research activities (**1**). Dendro provides a friendly web interface for humans as well as a series of APIs to enable other systems to manipulate files and folders as well as their metadata. Metadata creation is carried out using properties from different ontologies (either already present on the web or modeled by curators). With a triple store as the storage and querying layer, metadata can be added as property instances. Resources can also be retrieved using SPARQL queries, making faceted searches much easier to implement than on a relational model. Moreover, the simple triple store model enables external entities to easily query the data store via SPARQL.

2. Dataset deposit, where a set of files from Dendro, as well as their relevant metadata, are packaged and deposited in a long-term preservation platform such as Zenodo or CKAN(**2**)

3. Evolution of metadata recommendations (**3**). As the metadata specifications for different domains are created, they are also shared on the web, encouraging reuse and community-driven maintenance. Descriptor semantics become publicly documented and available for reuse in other data management systems, enabling a continuous evolution process that contributes towards the emergence of some ontologies as metadata standards for different research domains.

4. Data reuse (**4**). When a researcher accesses a dataset, documentation on the meaning of each descriptor will be available in the ontology from where that descriptor originated, making the interpretation of domain-specific metadata easier.

3. OVERVIEW OF THE PLATFORM

When designing the tools for an integrated preservation environment, one must ensure that the data stored within can survive the obsolescence of the environment itself. Dendro's triple-based data model, its reliance on shareable ontologies and a full open-source technology stack all contribute to maintaining access and interpretation of the stored datasets even after the platform's decommissioning.

Figure 1 shows the architecture of Dendro. The "Data" layer holds the data model for the platform, composed of three subsystems: an OpenLink Virtuoso Database (Open-Source version), an ElasticSearch server to enable distributed document indexing and a MongoDB/GridFS file storage cluster. The graph database is used to represent all the resources in the knowledge base (for example, `Researchers`, `Files`,

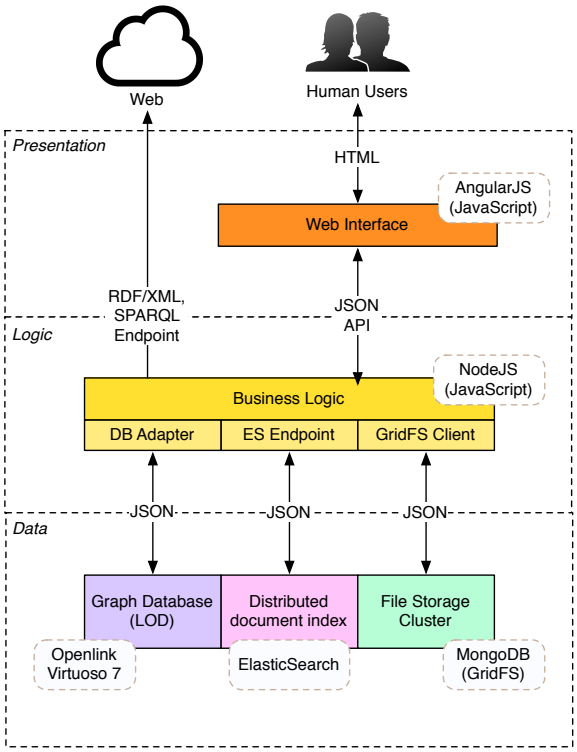

Figure 1: Dendro's architecture and technology stack

`Folders` and their attributes, represented using existing ontologies. Some of the ontologies being used at this time are Dublin Core Terms Ontology (for all resources in general), the Nepomuk File Ontology (for files and folder structures representation) and the Friend of a Friend Ontology (for describing platform `Users`). All queries specified by the *Logic* layer are sent to OpenLink Virtuoso's SPARQL endpoint. In case Virtuoso becomes obsolete, Dendro's triple-based model is designed to live on, since it can be fully exported in RDF and imported into another RDF-compliant solution. The triples plus the ontologies made available on the web enable a complete understanding of the stored information.

The *Logic* layer comprises Dendro's business logic, and includes three endpoints that connect to the underlying Data layer. A Database Adapter was written from scratch in order to provide a higher level of abstraction over the REST API provided by OpenLink Virtuoso. The module automatically performs the conversion between the results format provided by Virtuoso and Javascript objects to provide programmers an abstraction over the database, similar to Hibernate for Java or LINQ in the .NET platform.

The Logic Layer is written in NodeJS for handling large numbers of simultaneous connections—this allows numerous users or external systems (via Dendro's API) to interact directly with the platform to manage data and metadata. Dendro is primarily written in JavaScript, a simple and very widely known and used programming language among web developers—a plus when planning for an open-source preservation effort, as a large potential developer base makes main-

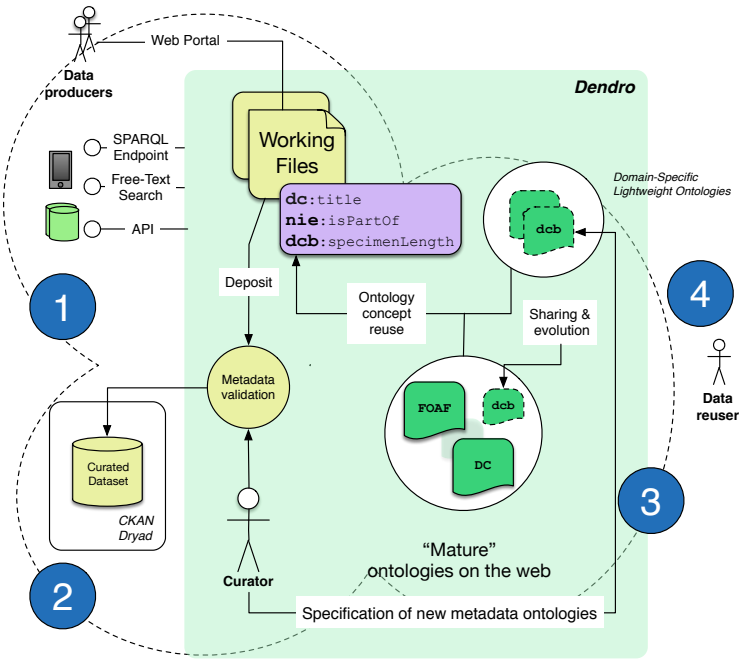

Figure 2: Dendro's role in a research data management ecosystem

tenance and evolution easier.

4. MANAGING DATASETS USING DENDRO

Figure 3 shows Dendro's main interface and the representation of recorded metadata in the triple store. Area **1A** shows the operations that can be performed over the current folder: Create a new folder, upload files, download the current folder, backup the current folder (includes metadata), restore a folder from a backup, and hide deleted files.

Area **1B** is the file explorer, showing the contents of the currently open folder. **1C** is a search box that allows any resource to be retrieved by any *literal* value (a continuously-updated index powered by ElasticSearch). **1D** exemplifies how domain-specific descriptors can be added to a metadata description; in this case, the `SpecimenLength` descriptor is added to the metadata for this folder. This descriptor has been previously specified in an ontology designed for mechanical engineering. Other descriptors from different ontologies can be loaded into the system, and the autocomplete box will retrieve them based on the values of their `rdfs:label` and `rdfs:comment` description properties. When a descriptor is selected by the user, it is added to the metadata editing area of the interface in the center. At the same time, the ontology from which it originates is "locked" so that the interface will suggest additional descriptors from the same ontology in a *quick-access* list of descriptors (Area **1E**). When a metadata value is inserted, it is recorded in the underlying triple store.

Area **2** shows a simple SPARQL query that obtains all the properties that have the folder being described as their subject. Although this is a very simple example, SPARQL allows resources in the knowledge base to be easily retrieved based on their properties and also on the properties of their linked resources. The results of the query are shown in (**3**)—note the descriptors from three different ontologies: *Dublin Core* (for generic metadata), *Nepomuk Information Element* (for file-related information) and *Double Cantilever Beam*, the domain-specific ontology for fracture mechanics datasets.

5. CONCLUSIONS

In this paper, we have presented Dendro, a collaborative research data management platform built on a triple store data model. Comparing it with repository platforms built on relational databases, we can see that the fully ontology-based data model provides a much more preservation-friendly environment, as it becomes self-documented. The *meaning* of the metadata values is specified in ontologies, which can evolve asynchronously according to the needs of different domains and be shared and retrieved from the web.

By representing datasets, papers, researchers and other research assets as resources and dataset metadata as values for properties relating these resources, a simple (triple-based) extensible (via ontologies) and powerful (supporting SPARQL querying) data model can be built.

Preliminary studies show that the platform satisfies several data management capabilities requested by researchers in our previous studies. We are now working on improving and testing it with researchers from different domains, while improving its interaction with existing repository platforms.

6. ACKNOWLEDGEMENTS

This work is supported by project NORTE-07-0124-FEDER-000059, financed by the North Portugal Regional Operational Programme (ON.2–O Novo Norte), under the Na-

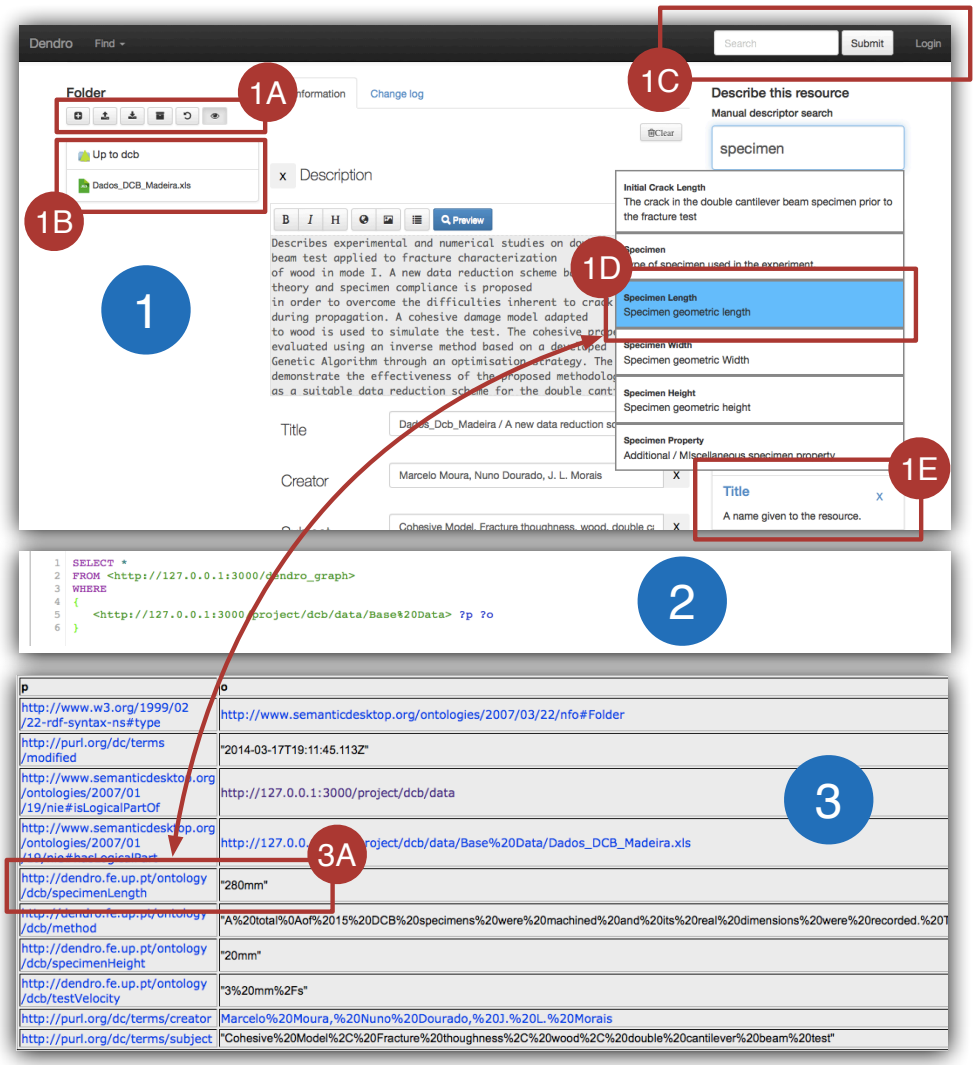

Figure 3: Dendro and its interaction with the triple store

tional Strategic Reference Framework (NSRF), through the European Regional Development Fund (ERDF), and by national funds, through the Portuguese funding agency, Fundação para a Ciência e a Tecnologia (FCT). João Rocha da Silva is also supported by grant SFRH/BD/77092/2011, provided by the Portuguese funding agency, Fundação para a Ciência e a Tecnologia (FCT).

References

Christine L. Borgman. 2012. The conundrum of sharing research data. *Journal of the American Society for Information Science and Technology* 63, 6 (2012). `http://ssrn.com/paper=1869155http://onlinelibrary.wiley.com/doi/10.1002/asi.22634/full`

Simon Hodson. 2011. *ADMIRAL: A Data Management Infrastructure for Research Activities in the Life sciences.* Technical Report. University of Oxford.

Yuan-fang Li, Gavin Kennedy, Faith Ngoran, Philip Wu, and Jane Hunter. 2013. An Ontology-centric Architecture for Extensible Scientific Data Management Systems. *Future Generation Computer Systems* 29, 2 (2013), 1–38.

Bryn Nelson. 2009. Data Sharing : Empty archives. *Nature* 461, September (2009). `http://europepmc.org/abstract/MED/19741679`

Open Knowledge Foundation. 2014. CKAN documentation - Release 2.2a. (2014).

João Rocha da Silva, José Barbosa, Mariana Gouveia, Cristina Ribeiro, and João Correia Lopes. 2013. UPBox and DataNotes: a collaborative data management environment for the long tail of research data. (2013).

David Shotton. 2012. The JISC UMF DataFlow Project : Introduction to DataStage. *Technical Report* (2012).